THE Exidy Monitor

August 1st, 1979 Volume I, Issue VI $1.00

Interfacing the
IBM Selectric TTY
-William Vetterling

Plane (Exidy Graphics II)
R. Rangel

NEW POLICIES

Michael Vachon

THE EXIDY MONITOR

printed monthly

John T. Vachon, Chief Editor
Thomas F. Harper, Software Consultant
Bruce R. McGlothlin, Coordinator
Marilyn Barry, Editorial Assistant

Cover art by Michael J. Vachon

Major conrtibutions by:

William T. Vetterling
R. Rangel

"The Exidy Monitor"
is sponsored by

COMPUTER MART OF MASSACHUSETTS
1395 MAIN STREET
WALTHAM, MA 02154
(617) 899-4540

Hello!

We're sorry for the delay in getting this issue to you, but we've been terribly busy reorganizing around here. However, this has enabled us to totally alter our filing system. This also starts our new policies for sending out issues, back-issues and volumes, which will be explained below.

First of all, all future volumes will be sent out on the first of each month, starting with this month. Subscriptions will be extended for the 15 day lapse so that everyone will receive 12 issues. Also, those who received the newsletter from issue 1 will be regarded as having started from issue 2 because of the size of the first issue. This means that you will receive 13 issues instead of 12.

Back-issues and software volumes are now distributed with the help of the computer, and a file of transactions will be kept. Issues will be sent out, along with the invoice, on or around the 1st and 15th of each month. If you have not received any back-issues or volumes yet that you requested before July 15, please let us know and we will get them to you as quickly as possible.

For those of you who are planning to send us a program, our policy is stated below. On receiving the program we will send out a general letter saying that we have received it and that it is being looked over. After our programmers review the program they will either find it acceptable or not acceptable. If they find it acceptable you will be fowarded a release to complete which will allow us to copyright the program to the User's Group. As specifically stated in the contract, this does not give us exclusive rights to the program; you are still free to sell it as you wish. This does, however, protect the program from unlawful duplication and distribution. If, however, the program is not found acceptable it will be returned to you with a brief explanation of why and how it might be improved.

When submitting programs, please make sure that they are well documented and include instructions on how to run the program. A brief description of what the program does will be appreciated if it is not evident from the program itself.

We have a few notices, also:

The first notice involves intelligent terminal software for the Sorcerer. We have had many requests for such software and would be very interested in including any software in the software library or even purchasing the software to make it available.

The second notice involves group meetings. At the time we are not prepared to organize and conduct such meetings, but there has been a general demand for them. We would, therefore, like to know if any of the members of the group would like to organize group meetings. If there is a large enough demand for this and if it is well organized we would be willing to let the meetings take place at Computer Mart, perhaps every other month. Let us know how you feel about this.

The third notice involves people who have recieved Development PACs but who have not recieved the manuals. We have been shipped manuals. We would like to verify which people this applies to. If you are waiting for the manual, please let us know.

Finally, we're asking once again for articles written by users. This newsletter can only be a success if people start contributing ideas and articles. We realize that many of you are still getting experienced with your Sorcerer and do not feel qualified to write an article. However, we would like to know what areas of application you are interested in and if you have had any problems.

****

We have a few corrections for last month's newsletter. These are corrections of lines in some of the programs that were included last month.

On page 4, in the INVERSE VIDEO program, line 240 should have read

```
240 PRINT CHR$(Z+128)
```

On page 9, in the BCHK program, line 20 should have read

```
20 B=32767-14*16^3
```

On page 11, in the WIDE PRINTER DRIVER routine (the BASIC program to enter the routine), lines 2020 and 2030 should have read

```
2020 POKE I, V
2030 NEXT I
```

****

July 3, 1979

Bruce McGlothlin
Computer Mart
1395 Main Street
Waltham, Mass.

Dear Bruce,

You suggested that it might be useful to write up the work I have done on interfacing an IBM selectric type terminal to the Sorcerer. As a point of interest, I am composing this letter with the Development Pac editor, and using the IBM selectric to type it out. Also, the listings enclosed were done in the same way.

The interface is a Solid State Music IO4 board. This board has the advantage that it shows in the instructions a very simple modification that will make the output run at 134.5 baud, the very unusual rate required by the selectrics.

My particular terminal is a GTE Novar 5-41, but many other varieties of selectric devices are available for about $600 to $700 dollars. Mine has an APL keyboard (which I don't use) and uses EBCD code. Some selectrics may be found with ASCII interfaces, and that makes things much more easy. However, the present listing shows how to handle the harder ones.

The basic idea is to intercept the ASCII code on its way out of the Sorcerer and to convert it to the proper form for whatever terminal you have. This involves setting up a code conversion table. A different table may be used for each individual typing ball. For example, I find that the Courier 72 ball is nice for letters like this, but when it comes to typing out BASIC programs, it is missing some of the important symbols. For BASIC I use the APL ball, which has only upper case characters, but lots of symbols.

The table is used in a simple fashion. Each ASCII code has a value between 0 and 127. Therefore, we make a table with 128 entries. Each entry is the code that we want to send out when we wish to type the corresponding ASCII character. For example, the ASCII code for a "0" is 48 (decimal), so the 48th entry in the table is the code we need to send out to get a "0" typed.

The only task left is to either find a listing of the codes or to create one. It is very simple in BASIC to find out what the appropriate codes are by experimenting. Simply use the command OUT 1,J and try values of J between 0 and 127. (I have assumed here that your terminal is connected to output port 1.)

You will find the device driver quite simple. It simply takes the ASCII code, adds it to the beginning address of the table, and uses the result as the address of the appropriate terminal code. There are a couple of finer details:

1) I adopted the convention in the table of using the highest order bit to indicate upper or lower case. The selectric has a separate codes for "shift up" and "shift down". If we are in lower case, and want to type a capital letter, we must send out two codes -- one to shift up, and one to type the character. Notice that this routine doesn't waste time shifting when it isn't necessary. When it shifts up, it stays up until a lower case character comes along, and the same is true of down-shifting.

2) After each character is sent out, a moment of silence is observed (by the routine DELAY) to allow the typing to occur.

3) In the selectric routine, there is a command to LD A,D. This command should only be used when the routine is running with the Development Pac. This Pac calls its output routines with the data in the D register. The monitor and BASIC have the data in the A register already, so in this case replace the code "7A" by a NOP code "00".

The code fits nicely in the range 00 to FF and can be used in several ways. The instructions following assume that you have loaded the code starting at location 00.

1) In the monitor, if you say "SE O=0F" the selectric becomes the output device and takes over for the screen. Then you can use it to dump memory as I have done here. Otherwise, you can say PP and get into BASIC where it can be used to produce a listing.

2) With the function USR(X) the terminal can be turned on and off directly from BASIC to produce hard-copy output. The technique for doing this was almost, but not quite, given in a previous newsletter. The actual USR(x) will be:

```
        LD A,00H
        LD DE,000FH           ;START ADDR OF HANDLER
        CALL E5F9H
        RET
```

and this code assembles to something like:

```
3E 00 11 0F 00 CD F9 E5 C9
```

This code can be put in the remainder of the space below FF and run by BASIC to start the output.

3) With the Development Pac the routine can be used as an output vector to produce source or assembler listings. For example, at present I am writing with the editor, and the IO vectors are set as follows:

```
        :CI     :SK
        :CO     :SV
        :0I     :AO
        :00     :AI
        :SI     :BI
        :SO     0F
```

Therefore, whenever I give a write command, or if I exit, the letter will come out on the terminal. This allows me to edit the letter so that you get a perfect copy.

The only thing I haven't mentionned is the routine INIT. It just so happens that my terminal won't do anything unless I begin my session by sending out a couple of particular initialization characters. This may or may not be necessary for other types of terminals. To use it, I load the entire code, and then use the monitor to say GO 00.

Sincerely,

William T Vetterling

William T. Vetterling

```
;
;
;************************************************************
;*                                                          *
;*  ROUTINE FOR INITIATING IBM SELECTRIC TERMINAL           *
;*  JULY 1,1979                                             *
;*                                                          *
;*  WILLIAM T. VETTERLING                                   *
;*                                                          *
;************************************************************
;
;
INIT    LD      A,0B4H
        OUT     (1),A
        CALL DELAY
        LD      A,0FCH
        OUT     (1),A
        CALL DELAY
        RET
;
;
;*********************************************************
;*                                                       *
;*     IBM SELECTRIC ROUTINE                             *
;*     JULY 2, 1979                                      *
;*     WILLIAM T. VETTERLING                             *
;*                                                       *
;*     ENTER WITH ASCII CODE IN "D" REGISTER             *
;*     ROUTINE LEAVES ALL REGISTERS UNCHANGED            *
;*     CODE CONVERSION TABLE IS AT "TABL"                *
;*                                                       *
;*********************************************************
;
;

        PUSH    BC      ;SAVE REGISTERS
        PUSH    DE
        PUSH    HL
        PUSH    AF
        LD      HL,TABL ;SET UP PTR
        LD      A,D     ;CODE TO "A"
        ADD     A,L     ;USE ASCII VALUE
        LD      L,A     ;  AS INDEX
        LD      A,(HL)
        LD      B,A     ;SAVE CODE
        LD      C,80H   ;CHECK SHIFT
        AND     C       ;NEW SHIFT POSITION
        LD      HL,FLAG
        LD      C,(HL)  ;GET OLD POSITION
        LD      (HL),A  ;STORE NEW ONE
        ADD     A,C      ;ADD OLD TO NW
        JP      Z,PRINT  ;IF DOWN,DOWN
        JP      C,PRINT  ;IF UP,UP
        LD      A,C      ;ELSE, CHANGE IT
        AND     A        ;IF OLD WAS DOWN,
        JP      Z,UP     ;  SHIFT UP
DWN     LD      A,1FH    ;ELSE SHIFT DOWN
        OUT     (1),A
        CALL DELAY
        JP      PRINT    ;THEN PRINT
UP      LD      A,1CH    ;SHIFT UP
        OUT     (1),A
        CALL DELAY
        JP      PRINT    ;THEN PRINT
PRINT   LD      A,B      ;GET STORED CHAR
        OUT     (1),A    ;OUTPUT
        CALL    DELAY
        POP     AF       ;RESTORE REGS
        POP     HL
        POP     DE
        POP     BC
        RET              ;RETURN
DELAY   LD      DE,-1    ;TIMING LOOP
        LD      HL,2200H         ;TIME=2200H CYCLES
LOOP    ADD     HL,DE    ;DECREMENT
        JP      C,LOOP   ;JUMP ON CARRY
        RET              ;RETURN
FLAG    NOP


;****************************************************
;*                                                  *
;*   CODE CONVERSION TABLE FOR IBM SELECTRIC        *
;*   COURIER 72 BALL              JULY 1, 1979      *
;*                                                  *
;*   WILLIAM T. VETTERLING                          *
;*                                                  *
;****************************************************
;
;

TABL    DEFW    0C0CH
        DEFW    0C0CH
        DEFW    0C0CH
        DEFW    0C0CH
        DEFW    0C1DH
        DEFW    0C0CH
        DEFW    2D0CH
        DEFW    0C0CH
DEFW    0C0CH
DEFW    0C0CH
DEFW    0C0CH
DEFW    0C0CH
DEFW    0C0CH
DEFW    0C0CH
DEFW    0C0CH
DEFW    0C0CH
DEFW    0100H
DEFW    0B089H
DEFW    8884H
DEFW    09A8H
DEFW    0A4B4H
DEFW    93B8H
DEFW    373BH
DEFW    0711H
DEFW    0624H
DEFW    3010H
DEFW    0804H
DEFW    2818H
DEFW    3438H
DEFW    2BABH
DEFW    1300H
DEFW    8700H
DEFW    0B990H
DEFW    0BAB6H
DEFW    8AAAH
DEFW    0A3B3H
DEFW    99A6H
DEFW    9A83H
DEFW    0A186H
DEFW    8592H
DEFW    9B8BH
DEFW    0A5A9H
DEFW    0B282H
DEFW    0B5B1H
DEFW    0A7A2H
DEFW    0A094H
DEFW    2007H
DEFW    0000H
DEFW    3909H
DEFW    3A36H
DEFW    0A2AH
DEFW    2333H
DEFW    1926H
DEFW    1A03H
DEFW    2106H
DEFW    0512H
DEFW    1B0BH
DEFW    2529H
DEFW    3202H
DEFW    3531H
DEFW    2722H
DEFW    0014H
DEFW    0000H
DEFW    0700H
```

00 for Monitor or BASIC
7A for DEV. PAC

```
DU 0 0FF

ADDR    0  1  2  3    4  5  6  7    8  9  A  B    C  D  E  F

0000:   3E B4 D3 01   CD 4E 00 3E   FC D3 01 CD   4E 00 C9 C5
0010:   D5 E5 F5 21   5A 00 00 85   6F 7E 47 0E   80 A1 21 59
0020:   00 4E 77 81   CA 43 00 DA   43 00 79 A7   CA 39 00 3E
0030:   1F D3 01 CD   4E 00 C3 43   00 3E 1C D3   01 CD 4E 00
0040:   C3 43 00 78   D3 01 CD 4E   00 F1 E1 D1   C1 C9 11 FF
0050:   FF 21 00 22   19 DA 54 00   C9 00 0C 0C   0C 0C 0C 0C
0060:   0C 0C 1D 0C   0C 0C 0C 2D   0C 0C 0C 0C   0C 0C 0C 0C
0070:   0C 0C 0C 0C   0C 0C 0C 0C   0C 0C 00 01   89 B0 84 88
0080:   A8 09 B4 A4   B8 93 3B 37   11 07 24 06   10 30 04 08
0090:   18 28 38 34   AB 2B 00 13   00 87 90 B9   B6 BA AA 8A
00A0:   B3 A3 A6 99   83 9A 86 A1   92 85 8B 9B   A9 A5 82 B2
00B0:   B1 B5 A2 A7   94 A0 07 20   00 00 09 39   36 3A 2A 0A
00C0:   33 23 26 19   03 1A 06 21   12 05 0B 1B   29 25 02 32
00D0:   31 35 22 27   14 00 00 00   00 07 00 00   00 00 00 00
00E0:   00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
00F0:   00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
```

```
LIST
 1 REM
 2 REM
 3 REM
 4 REM
 5 PRINTCHR$(12)
 6 GOSUB1000
 7 REM
 8 REM
 9 REM
 10 FORX=-3968TO-2112STEP64
 11 R=0
 12 REM
 13 REM
 14 REM
 15 REM
 16 REM
 20 POKEX,194
 21 REM
 22 REM
 23 REM
 24 REM
 25 REM
 30 POKE(X-64).32
 31 REM
 32 REM
 40 FORQ=1TO75:NEXTQ
 41 REM
 42 REM
 50 NEXTX
 51 REM
 52 REM
 53 REM
 60 FORQ=1TO50:NEXTQ
 61 REM
 62 REM
 63 REM
 64 REM
 65 REM
 70 POKE-2112,197
 71 REM
 80 FORQ=1TO50:NEXTQ
 81 REM
 82 REM
 83 REM
 90 POKE-2112,193
 91 REM
 100 FORX=-2112TO-2080
 101 REM
 102 REM
 110 POKEX,193
 111 REM
```

THIS AN EXAMPLE OF USER DEFINED GRAPHICS AND ANIMATION. IT DEFINES AN 8 BIT BLOCK ON THE SCREEN AND REDEFINES IT.

THIS WILL COUNT FROM THE UPPER TO THE LOWER SCREEN ADDRESS. WILL BE USED TO POKE GRAPHIC.

POKE INTO THE ADDRESS WE GOT AT LINE 10, THE FIRST USER DEFINED GRAPHIC IS 192, THE ONE WE WANT IS UNDER THE '3' KEY SO IT IS 194

NOW THAT WE HAVE POKED THE CHARACTER ONTO THE SCREEN, WE MUST GET RID OF THE OLD ONE. SO WE POKE 32 (SPACE) INTO THE SPACE WHERE THE OLD ONE IS.

OUR CHARACTER IS MOVING TOO FAST SO THIS WILL SLOW IT DOWN.

NOW DO TI AGAIN TO THE NEXT ADDRESS.

NOW WE WANT TO ROTATE THE PLANE TO FACE RIGHT. THIS IS A DELAY TO MAKE IT MOVE SLOWLY.

THE PLANE IS NOW FACING DOWN WE WANT TO PUT THE CHARACTER HERE THAT FACES DOWN AND TO THE RIGHT. SO WE POKE THAT CHARACTER IN THE -2112 (WHERE THE PLANE IS)

OH, NO! NOT ANOTHER DELAY LOOP!

NOW WE POKE INTO -2112 THE CHARACTER OF THE PLANE FACING RIGHT WHICH HAPPENS TO BE 193

NOW WE COUNT FROM LEFT TO RIGHT.

AND POKE 192 (PLANE FACING RIGHT INTO THE COUNT AT LINE 100

POKE A SPACE OVER THE OLD PLANE

```
112 REM                               AND AWAY WE GO
120 POKE(X-1),32
121 REM                               IT'S DELAY TIME AGAIN!
130 FORQ=1TO75:NEXTQ
131 REM                               DO IT AGAIN UNTIL WE GET TO THE
132 REM                               LOWER RIGHT HAND CORNER.
133 REM                               WE DO THIS OVER AND OVER UNTIL
134 REM                               WE GET BACK TO THE BEGINING.
140 NEXTX
150 FORQ=1TO50:NEXTQ
160 POKE-2080,196
170 FORQ=1TO50:NEXTQ
180 POKE-2080.192
190 FORX=-2080TO-3936STEP-64
200 POKEX,192
210 POKE(X+64),32
220 FORQ=1TO75:NEXTQ
230 NEXTX
240 FORQ=1TO50:NEXTQ
250 POKE-3936,199
260 FORQ=1TO50:NEXTQ
270 POKE-3936,195
280 FORX=-3936TO-3968STEP-1
290 POKEX,195
300 POKE(X+1),32
310 FORQ=1TO75:NEXTQ
320 NEXTX
330 FORQ=1TO50:NEXTQ
340 POKE-3968,198
350 FORQ=1TO50:NEXTQ
351 POKE-3968,194:IFR=1THEN10
352 FORQ=1TO50:NEXTQ
353 POKE-3968,197
354 FORQ=1TO50:NEXTQ
355 POKE-3968,32
358 R=R+1
359 Y=0
360 FORX=-3968TO-2080STEP64
370 Y=Y+1,
380 POKE(X+Y),197
400 FORQ=1TO75:NEXTQ
405 POKE(X+Y),32
410 NEXTX
420 GOTO190
500 REM                               THIS IS THE ARRAY THAT WILL BE
501 REM                               POKED INTO THE USER DEFINED
502 REM                               GRAPHIC SPACE IN MEMORY.  THE
503 REM                               HEX VALUE FOR EACH LINE OF THE
504 REM                               CHARACTER IS CONVERTED TO DECIMAL
505 REM                               AND IS POKED INTO MEMORY IN THE
506 REM                               PROPER PLACE.  THIS ALLOWS THE
```

```
LIST
507 REM                      PROGRAM TO BE ENTIRELY IN BASIC
600 REM
610 REM
620 REM                (I HAVE RE-WRITTEN THIS SECTION OF THE
630 REM                 PROGRAM SO THAT INSTEAD OF HAVING 63 LINES
640 REM                 THAT EACH DEFINED ONE PART OF THE ARRAY
650 REM                 THAT WAS TO BE USED TO POKE IN THE
660 REM                 GRAPHICS, I HAVE DONE AWAY WITH THE ARRAY
670 REM                 TOTALY AND AM READING IN THE VALUES FROM
680 REM                 DATA STATEMENTS AND POKEING THEM INTO
690 REM                 USER GRAPHIC RAM IMMEDIATELY.
700 REM
710 REM                              --JOHN VACHON)
720 REM
1000 FOR A=0 TO 63
1010 READ GR
1020 POKE A, GR
1030 NEXT A
1040 RETURN
1050 REM           (NOW THE DATA...)
2000 DATA 0, 8, 127, 127, 8, 8, 28, 0, 0, 12, 12, 76, 126
2010 DATA 76, 12, 12, 0, 56, 16, 16, 126, 126, 16, 0, 48, 48
2020 DATA 50, 126, 50, 48, 48, 0, 16, 40, 28, 14, 87, 34, 16
2030 DATA 0, 0, 16, 34, 87, 14, 28, 40, 16, 0, 4, 66, 229
2040 DATA 112, 48, 24, 8, 8, 24, 48, 112, 229, 68, 4, 0
5000 REM
5001 REM
5002 REM                        EXIDY INC.
5003 REM                          CUSTOMER SERVICES DEPT.
5004 REM                           JAN. 15, 1979
5005 REM                                        R. RANGEL
READY
```

Sorcerer User's Group
Software Library Volumes

VOLUME	FILE	MEMORY	DESCRIPTION OF PROGRAM
VOL I	AMAZE	8K	DRAWS MAZES ON THE SCREEN
	LUNAR	8K	LUNAR LANDER SIMULATOR
	ROTE	8K	LETTER-ORDERING GAME
VOL II	LUCAS	8K	SETS UP "LUCAS' PUZZLE"
	RSP	8K	GAME OF ROCK, SCISSORS, PAPER
	CHECK	8K	SOLITARY CHECKERS GAME
VOL III	DEPTH	8K	DEPTH-CHARGE GAME WHERE YOU SINK THE TRS-80
	REVRS	8K	NUMBER REVERSING GAME
	HIQUE	8K	PLAYS THE ANCIENT GAME "HI-Q"
VOL IV	BIO	8K	GIVES A BIO-RHYTHM CHART
	MIDAG	8K	LETS YOU CONTROL A MIDDLE-AGES MANOR
	MANDL	8K	PRINTS A MANDALA ON THE SCREEN
VOL V	HANGM	16K	HANGMAN PROGRAM W/ 173 WORDS
	HPAWN	16K	PLAYS HEXAPAWN
VOL VI	HANG2	32K	HANGM W/ 600 HOUSEHOLD WORDS, 50 PHRASES
	HANG3	16K	HANGM W/ 333 HOUSEHOLD WORDS, 50 PHRASES
VOL VII	WARPX	32K	STAR TREK W/ SOUND EFFECTS THROUGH MUSIC BOARD
	TREK1		INSTRUCTIONS FOR STAR TREK
VOL VIII	WARPZ	32K	STAR TREK W/ SOUND EFFECTS THROUGH RMP INDUSTRIES MUSIC BOX
	TREK1		INSTRUCTIONS FOR STAR TREK
VOL IX	BANER	*	PRINTS UP BANNERS FOR THE SCREEN OR PRINTER
	BOXOP	8K	EXPERIMENTAL MATH TRIAL-AND-ERROR DEMO
VOL X	DEMO	32K	EXIDY 32K DEMONSTRATION PROGRAM
	CHAR	8K	CHARACTER GENERATOR PROGRAM
VOL XI	DUMBT	8K	DUMB TERMINAL PROGRAM
	DUMB2	8K	DUMB TERMINAL PROGRAM FOR IBM AND DEC

* Requires 16K to do banners in small size, 32K to do banners in larger size letters.

Volumes: $9.99 for non-User Group Members
$6.50 for User Group Members

SORCERER SYSTEMS AND PERIPHERALS
AVAILABLE FROM COMPUTER MART, INC.

8K Sorcerer	$ 949.00
8K Sorcerer w/ Leedex Video 100 Monitor & video cable	1085.00
16K Sorcerer	1145.00
32K Sorcerer	1285.00
32K Sorcerer w/ Leedex Video 100 Monitor & video cable	1395.00
Leedex Video 100 12" Monitor	149.00
S-100 Expansion Cabinet w/ power supply	349.00
Sorcerer Video cable	15.00
Integral Data Systems IP-125 Printer w/ Printer control	838.00
Integral Data Systems 440 Printer w/ Forms control	995.00
Graphics and 2K buffer options for the 440	199.00
Other Printers (See our price list for printers)	
Sorcerer Cable for IDS Printers	50.00
Exidy S-100 Expansion Interface	199.00
32K Memory Expansion Kit	219.00
Installed and Guaranteed	339.00
Sorcerer Operations Manual	9.95
Sorcerer BASIC manual, Technical manual, Word Processing manual, or Development Tour manual; each:	11.95
Sorcerer Standard BASIC ROM PAC, Development ROM PAC, Word Processor ROM PAC each:	99.00
Sorcerer EPROM PAC	49.00
Serial Data Cassette Cable	75.00
D-30 Data Cassettes	1.50
Data Cassette Recorder	55.00
Tek-Com Accoustic Coupler Modem TC-3006 (1 yr warranty)	215.00
RMP Industries Music Box for Sorcerer	49.95
Vista Mini-Disk System w/ Disk BASIC E and CP/M v. 1.3	599.00
Extra Drive, Cabinet and Power Supply	475.00
Micropolis DD Mod I disk System w/ Disk BASIC and MDOS	795.00
Additional drive w/ cabinet and power supply	545.00
Micropolis QD Mod II Disk System w/ Disk BASIC and MDOS	1145.00
Dual Drive Mod II	1895.00
CP/M for Micropolis w/ Sorcerer Drivers	145.00
Micromation Doubler Interface w/ dual drive cable	539.00
CP/M w/ BASIC E for Doubler w/ Sorcerer drivers	125.00
Extra 8 inch Drive w/ Cabinet and Power Supply	795.00
Micromation MEGABOX for Sorcerer w/ CP/M and BASIC E	2295.00
Single drive MEGABOX system	1595.00
One year warranty extension	150.00
Full line of S-100 Boards (See our S-100 price list)	
Sorcerer Software on Cassette	
Programma International (several packages)	9.95
GRT "Clinic," "Beat the House," etc.	14.95
Creative Computing "Graphic Games-2"	7.95
Basically Speaking Software (several packages)	11.95
Sorcerer User's Group Volumes	6.50
Sorcerer Software on Diskette (Specify type of Disk System)	
CCA Data Management (requires Micropolis BASIC)	150.00
Compumax Business System (req. Micropolis BASIC)	500.00
Microsoft BASIC (requires CP/M)	350.00
Microsoft Macro-80 (requires CP/M)	149.00
Microsoft Fortran (requires CP/M)	500.00
Microsoft COBOL (requires CP/M)	750.00
Xitan Disk BASIC (requires CP/M)	159.00
Xitan Disk Z-TEL Text Editor, Text Output Processor, Macro Assembler or Linker (req. CP/M); each:	69.00
Xitan Disk Fortran IV (requires CP/M)	349.00
Xitan Disk Z-BUG Debugger (requires CP/M)	89.00
Xitan Disk Z-TEL, MACROASM, Text Output Processor and SUPER BASIC (requires CP/M)	249.00
C BASIC II (requires CP/M)	95.00
Structured Systems Q SORT (requires C BASIC)	95.00
Selector III-C2 Data Base Management (r. C BASIC)	345.00

Prices effective July 25, 1979

"The Exidy Monitor"
back-issues

VOLUME I:

[ ] Issue 1:	$ .25:	printer control, useful routines
[ ] Issue 2:	1.00:	PEEK, POKE, USR, memory test
[ ] Issue 3:	1.00:	D-PAC, Z-80 I, Misc info, questionnaires
[ ] Issue 4:	1.00:	Z-80 II, Software library, changes
[ ] Issue 5:	1.00:	Wide Printer driver, RND, INVRS, graphics, TESTS

SOFTWARE VOLUMES

Volume	Contents	Price
I	AMAZE, LUNAR, ROTE	[ ] $6.50
II	LUCAS, RSP, CHECK	[ ] 6.50
III	DEPTH, REVRS, HIQUE	[ ] 6.50
IV	BIO, MIDAG, MANDL	[ ] 6.50
V	HANGM, HPAWN	[ ] 6.50
VI	HANG2, HANG3	[ ] 6.50
VII	WARPX, TREK1	[ ] 6.50
VIII	WARPZ, TREK1	[ ] 6.50
IX	BANER, BOXOP	[ ] 6.50
X	DEMO, CHAR	[ ] 6.50
XI	DUMBT, DUMB2	[ ] 6.50

(enter amount wanted of each in brackets)

NAME: ____________________
ADDRESS: ____________________
CITY: ____________ STATE: ____________ ZIP: ________

TOTAL PURCHASE: $ ________
Plus Shipping: $ 1.00
MA residents add 5% sales tax: ________ (only for Volumes)

TOTAL: $ ________

Enclosed is $______ for the above checked off volumes and/or back-issues.

[ ] CASH, CHECK, or MONEY ORDER
($25.00 minimum on charges:)
[ ] VISA/BANK AMERICARD [ ] MASTERCHARGE
Card # Exp. date: ____________________